Quaestiones Informaticae

An official publication of the Computer Society of South Africa
'n Amptelike tydskrif van die Rekenaarvereniging van Suid-Afrika

Editors: Dr. D. S. Henderson,
Vice Chancellor, Rhodes University, Grahamstown, 6140, South Africa.
Prof. M. H. Williams,
Department of Computer Science and Applied Maths,
Rhodes University, Grahamstown, 6140, South Africa.

Editorial Advisory Board

PROFESSOR D. W. BARRON
Department of Mathematics
The University
Southampton S09 5NH
England

MR. P. P. ROETS
NRIMS
CSIR
P.O. Box 395
PRETORIA 0001
South Africa

PROFESSOR S.H. VON SOLMS
Department of Computer Science
Rand Afrikaans University
Auckland Park
Johannesburg 2001
South Africa

PROFESSOR K. GREGGOR
Computer Centre
University of Port Elizabeth
Port Elizabeth 6001
South Africa

PROFESSOR G. WIECHERS
Department of Computer Science
University of South Africa
P.O. Box 392
Pretoria 0001
South Africa

PROFESSOR K. MACGREGOR
Department of Computer Science
University of Cape Town
Private Bag
Rondebosch 7700
South Africa

MR. P. C. PIROW
Graduate School of Business Administration,
University of the Witwatersrand
P.O. Box 31170
Braamfontein 2017
South Africa

PROFESSOR G. R. JOUBERT
Department of Computer Science
University of Natal
King George V Avenue
Durban 4001
South Africa

Subscriptions

Annual subscriptions are as follows:

<table>
<thead>
<tr>
<th></th>
<th>SA</th>
<th>US</th>
<th>UK</th>
</tr>
</thead>
<tbody>
<tr>
<td>Individuals</td>
<td>R2</td>
<td>$3</td>
<td>£1.50</td>
</tr>
<tr>
<td>Institutions</td>
<td>R4</td>
<td>$6</td>
<td>£3.00</td>
</tr>
</tbody>
</table>

Quaestiones Informaticae is prepared for publication by SYSTEMS PUBLISHERS (PTY) LTD for the Computer Society of South Africa.
Teacher Control in Computer-Assisted Instruction

Peter Calingaert

University of North Carolina at Chapel Hill
Department of Computer Science
New West Hall 035-A, Chapel Hill, NC 27514, USA

Abstract
To many persons, computer-assisted instruction (CAI) connotes automated programmed instruction (PI). This exemplifies the author-controlled mode of CAI, in which PI, suitably generalized, is mediated by the computer. This mode is used effectively for tutorial instruction. In another mode of CAI, the interaction between student and computer is under control of the student, perhaps within bounds set by an author. This student-controlled or discovery mode is most often seen in problem solving and in simulation, such as the ersatz laboratory. A third mode, teacher-controlled CAI, is assuming increasing importance. Here the teacher intervenes in the student-computer interaction, generally with an entire class watching one dialog. In the hands of a skillful teacher, this mode offers most of the pedagogical advantages of the other two modes and costs far less.

1. Introduction
Many ways of using the computer in instruction have been explored. Indeed, the expression computer-assisted instruction (CAI) has acquired almost as many meanings as users. Rather than attempt a precise definition, I shall simply place a fairly broad construction upon the expression, encompassing by it most situations in which a computer system is interacting with one or more students who are attempting to derive knowledge or understanding from the system. I have found it enlightening to classify the many resulting forms of CAI according to the agent controlling the interaction between student and computer.

At one extreme, that agent may be the author of instructional materials being mediated by the computer. At the opposite extreme is the student himself, deciding which programs (including his own) are to be invoked, in what sequence, and with what parameters. A middle ground is represented by the student’s teacher as the agent of control. The boundaries separating these three modes of CAI are, of course, not always sharp. The essential differences which nevertheless distinguish them will be illustrated by several examples.

2. Author Control

Genesis
Author-controlled CAI is a direct descendant of programmed instruction (PI). The branching form of PI, due to Crowder [7], permits the individualization of subject matter. It is often encountered in the form of a scrambled PI text. In this form, a multiple-choice question is asked of the student, and his choice of answer dictates the next selection of material to be presented to him. The linear form of PI traces its mechanization to Pressey [16, 17] and its defense to Skinner [18], who emphasized the importance to educational effectiveness of having the student construct rather than choose the response to a question. To use a constructed response for branching requires automatic answer analysis. Not having this, Skinner preferred to keep constructed responses and gave up branching and therefore individualization. The difficulty of analyzing constructed responses by computer is responsible for the lesser impact which linear PI has had on CAI than has branching PI.

The obvious branching capability of the computer has made it a natural vehicle for the administration of branching PI. This may well be termed computer-administered programmed instruction (CAPI). In a computer, the branching can depend (in theory if only occasionally in practice) not only upon the choice of response, but upon the student’s last several responses, the time he takes to enter a response, his age, or a pseudorandom number. Moreover, the computer can record the interactions which occur and prepare analyses of student performance and of question performance. The erasability of the computer’s storage media permits one to modify a set of instructional materials without reprinting an entire book or filmstrip. Frequent evolutionary improvement is therefore quite feasible.

Examples
The most glamorous form of CAPI is tutorial instruction in which the computer system presents new material to the student in a sequence and at a pace tailored to his individual needs. The multiple learning tracks available are selected by student responses to interspersed diagnostic questions. It may be argued that the interaction in tutorial instruction is under control of the student, because the actions of the computer are dictated solely by the student’s responses. The student does not
know, however, what his responses signify to the computer. It is the course author who has determined for each possible student response what the corresponding computer action will be. The interaction is truly under author control.

The questions used to guide tutorial instruction may be effective in the absence of the tutorial material. If no instruction is contemplated, this is computer-administered testing. With the tutorial material retained, various forms of CAI are possible. If the questions test the student’s command of facts, such as the multiplication table, the computer is administering drill. If the questions test the student’s application of facts and techniques, as in multiplying two-digit numbers, then the computer is administering practice. If the questions are those on which a student has worked in advance, perhaps from a textbook, the computer is administering recitation, or problem review.

**Evaluation**

Author-controlled CAI has the following advantages.

1. It works: CAPI can be as effective as conventional classroom instruction [19].
2. It permits individualized instruction.
3. It affords privacy for making mistakes. This is particularly important for adults.
4. It allows flexibility of scheduling, but does not demand flexibility unless the number of students in a class exceeds the number of terminals available.

Many disadvantages, however, must be considered.

1. It is rather expensive to deliver instruction this way. For estimates ranging from very optimistic to very pessimistic, see [3, 6, 9, 14].
2. It is very expensive to develop CAPI materials [3, 6, 9, 14]. Estimates of the order of 100 hours of professional work per hour of lesson have been made since the mid-1960’s and are still common as the 1980’s approach. Effective the instruction may be, but it is rarely cost-effective.
3. If constructed responses are permitted, response analysis is difficult and determination of what action to take in return is equally difficult.
4. If constructed responses are not permitted, the instruction is severely constrained and teaching strategies are limited. Many CAPI materials are consequently monotonous.

**3. Student Control**

**Forms**

Student-controlled CAI gives the student a more active role in determining how the computer is to assist in instructing him. Many forms of this mode of CAI can be envisaged. A particularly exciting form is the automated library, which the student uses from his terminal. The problem of browsing has been mitigated, although not solved, by the development of networks of textual information, such as Nelson’s hypertexts [13]. Despite continued advances in the development of question-answering systems, however, the many problems of storing and retrieving information, whether in the form of documents or of facts, make this type of student-controlled CAI still more a wish than a reality. Mention of the use of a computer to carry out a student’s computation is perhaps obvious. It belongs in the catalog, nevertheless, because it is common and effective. An especially attractive form of this use is provided by the hand-held electronic calculator, with its rich set of primitive functions. The even richer set offered by a language such as APL creates not merely a quantitative, but rather a qualitative difference between modern computer and old-style calculator. Moreover, the availability of program packages, such as for statistical analysis, further enhances the applicability of the computer.

Despite the foregoing, the notion of student-controlled CAI most often suggests experimentation with models. If the model is built by an author, the experimentation is called simulation, and the student is supposed to learn not by instruction but by discovery. Models have been written for such subjects as mathematics, chemistry, and economics [10]. Often called ersatz laboratories (a term apparently due to Adams [1, 2]), these models can be used when the true laboratory would be too expensive, too dangerous, or too time-consuming. Moreover, simplifying assumptions (e.g. absence of friction) and random variations (e.g. measurement errors) can be inserted or withheld to the degree desired.

If the model is built by the student, the experimentation is called problem solving. Teachers recognize that while teaching they also learn. A student, too, can learn while teaching, given a suitable pupil. Such a pupil is provided by the computer. The need for an algorithm forces the student to develop a logically consistent method of problem solution.

If models of the same phenomenon are prepared by both author and student, the computer can compare the models. By invoking the student’s model with suitably chosen parameters, the computer system can perform diagnostic checking by looking for certain classes of errors in the student’s model [12]. Alternatively, the student himself can compare the behavior of his model with the behavior of the computer’s model [8]. It is possible, of course, to limit the checking to simple comparison of outputs without diagnosis. A variety of the latter, common in arithmetic drill, is for the student and computer each to pose problems to the other, with the student deciding for each successive problem which partner is to set the problem and which to solve it.

**Evaluation**

Student-controlled CAI shares the advantages previously cited for author-controlled CAI. It works [4, 5]; instruction is individualized; interaction is private; scheduling is flexible. Further advantages peculiar to the student-controlled mode are the following.

1. The student can receive a meaningful and pertinent answer to a response which was not specifically anticipated by the author.
2. The student’s involvement in the interaction is active, not passive. This is likely to increase his motivation to learn [11].
3. Many types of materials, such as the ersatz laboratory, cannot reasonably be presented in author-controlled mode.

The usual cost disadvantages are present for student-controlled as for author-controlled CAI, although it is possible that the ersatz laboratory can be less costly than the real laboratory. A major disadvantage of student-controlled CAI is that supervision of the student becomes much more difficult. The author cannot keep the student from straying. If a teacher is present, part of the potential privacy is lost, and it is not clear to what extent the teacher should attempt to guide the student, nor even how best to guide him. My colleagues and I have had good experience, however, with a written laboratory instruction which the student could follow at his option if no better ideas struck him.

A further characteristic of this mode may be either an advantage or a disadvantage. It is very often the student who decides whether his understanding is right or wrong.

**4. Teacher Control**

**Philosophy**

Teacher-controlled CAI represents an attempt to capture some of the advantages of the other two modes, without their disadvantages. The attempt rests on two complementary strategies. The first is to
make use of the teacher, who, unlike the author, is present and, unlike the student, has the relevant knowledge, experience, and judgement to manage the student's interaction with the computer. The second is to treat the computer system not as a master but as a tool, much the way the teacher customarily treats chalkboard, textbook, and projector.

**Mechanism**

One tactic for achieving teacher control is to insert the teacher into an author-student system with means for monitoring and overriding the interactions [5]. This mechanism, commonly used in language laboratories, is not unlike the modes already discussed, with added complications. It is potentially more powerful, but surely still very costly.

A less expensive tactic is to place one computer terminal in the classroom, where it is operated by the teacher. The one dialog is made visible to the several students by means of opaque overhead projector, closed-circuit television, or directly-connected video monitors. This second tactic is much simpler than the former, and deserves further scrutiny.

**Examples**

Two topics which I have taught in this mode are the iterative solution of systems of linear equations and the design of a serial binary adder. Although my personal interests have led me to select topics with considerable mathematical content, such is no more a prerequisite for teacher-controlled than for author- or student-controlled CAI. What is actually required is that the topics to be taught have as much structure as is needed to make them suitable for the student-controlled mode.

I introduce the first topic by displaying, using chalkboard or overhead projector, an arbitrary \( n \) by \( n \) system of equations written first in the customary manner, and then in a form in which the \( i \)th equation has been solved for the \( i \)th unknown. I then turn to a computer terminal whose output is visible to the class over closed-circuit television. Programs have been stored for checking a convergence criterion, for solving a linear system by the Seidel method (of successive displacements), and for solving one by the Jacobi method (of simultaneous displacements). The latter two programs accept the initial approximation as a parameter and print out every \( j \)th iterate, where \( j \) can be reset at any time.

By using these programs, written in a language which need not be familiar to the students, they can explore with me the effect on convergence (which is truly seen as a process) of the choices of method, of starting value, and of order of the equations. Although I start with a preplanned sequence of such choices, I usually modify them as I gauge my students' understanding and, indeed, in response to their questions and suggestions. The interaction which I supervise between the computer and the students can thus contain pedagogical elements found in the tutorial, desk calculator, simulation, and problem-solving forms of CAI. The lesson continues with a non-computer explanation of linear acceleration, followed by the use of a computer program based on the Seidel method, with acceleration used to calculate every \( k \)th iterate.

The binary adder presentation builds up computer program models of a half adder, a full adder, and a serial adder. Block diagrams of these devices are also shown. For this particular unit of instruction it is necessary that the students be able to program in the language used, because they are called upon to help rewrite an erroneous model of the serial adder.

We have also used a graphic display system programmed to illustrate numerical methods of solving nonlinear equations and ordinary differential equations. This has been shown, not surprisingly, to be clearly more effective than spending an equal amount of time in conventional classroom instruction [15].

**Evaluation**

Teacher-controlled CAI, as here envisaged, forgoes three of the advantages of the other two modes. Most of the individual attention is lost, as well as privacy and flexibility of scheduling. Computer system failure now discommodes a whole class at once rather than an individually scheduled student. The teacher must therefore have a backup plan. This may be either an alternative presentation of the same material or a presentation of other material which is not enhanced by the computer. Of course a backup plan is an asset for the other modes of CAI also.

Teacher-controlled CAI has been shown to work, however, and I feel that the foregoing disadvantages, all minor save the loss of individualization, are outweighed by the following advantages.

1. Relying on his perception of the understanding and needs of his particular students, the teacher can effectively filter student inputs. In reply to a question, he can (a) transmit it unchanged to the computer, (b) rephrase and then transmit it, (c) request the student to rephrase it, (d) reject it as frivolous, (e) encourage the student to determine the answer without the computer, etc.

2. The teacher can interleave, with whatever frequency he judges best at the moment, prepared demonstrations and *ad hoc* responses to student questions.

3. The analysis of constructed responses and questions posed by the students is performed by a highly intelligent subsystem, the human teacher.

4. The teacher can use materials prepared by an author for the student-controlled mode and, with additional effort, materials for the author-controlled mode. He is free, of course, to modify such materials or to prepare his own.

5. Teacher-controlled CAI is *much cheaper* than the other two modes. A large reduction in development costs ensues because the teacher can be expected to handle exceptional situations which must otherwise be provided for in the computer program. Moreover, all terminal and communication costs and many computer system costs are divided by the class size, as compared with the terminal-per-student situation.

**5. Conclusion**

For over a decade there has been no doubt of the effectiveness of various modes of CAI. Cost-effectiveness, on the other hand, has too often been ignored, as many CAI projects have been funded and judged chiefly as components of research. Viewed more narrowly as components of an instruction delivery system, they must provide educational benefits commensurate with their costs. I have argued here that the replacement of author control and student control by teacher control is a major step toward reduction of the costs.

It may be objected that only a particularly skillful teacher can make effective use of such a complex and versatile tool as the computer. The ineffective manner in which some teachers use the chalkboard or textbook lends credence to that argument. Nevertheless, just as we attempt to teach prospective teachers not to erase what they have just written, nor to read long passages *verbatim* from the textbook, so can we attempt to teach the proper use of the computer as a tool. *Any* form of teaching is best in the hands of a skillful teacher.

Costs will have to be reduced yet further for CAI to be as attractive in primary and secondary as in post-secondary education. The extra time required can be well utilized in exposing future primary
and secondary teachers to the uses of the computer in education. It should be stressed that each teacher who wishes to enhance his teaching by use of a computer need not be compelled to develop his own materials. It is quite possible for authors to prepare CAI materials with teacher control in mind. There exist problems of standardization and distribution, to be sure, but these are no more severe than for the other modes of CAI.

The author-controlled, student-controlled, and teacher-controlled modes all use the capabilities of the computer system in different manners. Teacher-controlled CAI offers most of the pedagogical advantages of the other two modes and costs far less. It therefore merits not only further investigation but also more widespread implementation.

References

Notes for Contributors

The purpose of this Journal will be to publish original papers in any field of computing. Papers submitted may be research articles, review articles, exploratory articles or articles of general interest to readers of the Journal. The preferred languages of the Journal will be the congress languages of IFIP although papers in other languages will not be precluded.

Manuscripts should be in double-spaced typing on one side only of A4 paper and submitted to Dr. D. S. Henderson or Prof. M. H. Williams at
Rhodes University
Grahamstown 6140
South Africa

Form of manuscript

Manuscripts should be in double-space typing on one side only of sheets of A4 size with wide margins. The original ribbon copy of the typed manuscript should be submitted. Authors should write concisely.

The first page should include the article title (which should be brief), the author's name, and the affiliation and address. Each paper must be accompanied by a summary of less than 200 words which will be printed immediately below the title at the beginning of the paper, together with an appropriate key word list and a list of relevant Computing Review categories.

Tables and figures

Illustrations and tables should not be included in the text, although the author should indicate the desired location of each in the printed text. Tables should be typed on separate sheets and should be numbered consecutively and titled.

Illustrations should also be supplied on separate sheets, and each should be clearly identified on the back in pencil with the Author's name and figure number. Original line drawings (not photoprints) should be submitted and should include all relevant details. Drawings, etc., should be about twice the final size required and lettering must be clear and "open" and sufficiently large to permit the necessary reduction of size in block-making.

Where photographs are submitted, glossy bromide prints are required. If words or numbers are to appear on a photograph, two prints should be sent, the lettering being clearly indicated on one print only. Computer programs or output should be given on clear original printouts and preferably not on lined paper so that they can be reproduced photographically.

Figure legends should be typed on a separate sheet and placed at the end of the manuscript.

Symbols

Mathematical and other symbols may be either handwritten or typewritten. Greek letters and unusual symbols should be identified in the margin. Distinction should be made between capital and lower case letters between the letter O and zero; between the letter l, the number one and prime; between K and kappa.

References

References should be listed at the end of the manuscript in alphabetical order of author's name, and cited in the text by number in square brackets. Journal references should be arranged thus:


Proofs and reprints

Galley proofs will be sent to the author to ensure that the papers have been correctly set up in type and not for the addition of new material or amendment of texts. Excessive alterations may have to be disallowed or the cost charged against the author. Corrected galley proofs, together with the original typescript, must be returned to the editor within three days to minimize the risk of the author's contribution having to be held over to a later issue.

Fifty reprints of each article will be supplied free of charge. Additional copies may be purchased on a reprint order form which will accompany the proofs.

Only original papers will be accepted, and copyright in published papers will be vested in the publisher.

Letters

A section of "Letters to the Editor" (each limited to about 500 words) will provide a forum for discussion of recent problems.

Hierdie notas is ook in Afrikaans verkrygbaar.
Questiones Informaticae


Contents/Inhoud

Toepaslikheid van 'n analitiese model by die keuse van 'n leerstruuktur vir 'n teksverwerkingstelsel ....... 1
A. Penzhordn

Direct FORTRAN/IDMS interface (without the use of a DML) on the ICL 1904/2970 computers, using a geological data base as example ....................................................... 12
B. Day

Rekenaarondersteunde onderhoud van programmatuurstelsels. ....................................................... 16
E. C. Anderssen

Database design: choice of a methodology ................................. 23
M. C. F. King, G. Naudé, S. H. von Solms

Design principles of the language BPL ................................. 26
M. H. Williams

A high-level programming language for interactive lisp-like languages ................................. N/A
S.W. Postma

The sequence abstraction in the implementation of EMILY ................................. 30
D. C. Currin, J. M. Bishop, Y. L. Varol

Block-structured interactive programming system ................................. N/A
C. S. M. Mueller

The management of operating systems state data ................................................. 34
T. Turton

From slave to servant. ................................................. N/A
G. C. Scarrott

Teacher control in computer-assisted instruction ................................................. 38
P. Calingaert

The impact of microcomputers in computer science ................................................. N/A
K. J. Danhof, C. L. Smith

Architecture of current and future products ................................................. 41
C. F. Wolfe

An algorithm for merging disk files in place ................................................. 44
P. P. Roets

An algorithm for the approximation of surfaces and for the packing of volumes ................................................. 44
A. H. J. Christensen

The memory organisation of large processors .................................................................. N/A
D. M. Stein